

Modeling the Temperature Bias of Power Consumption for Nanometer-Scale CPUs in Application Processors

Karel De Vogeleer*, Gerard Memmi*, Pierre Jouvelot† and Fabien Coelho†

*TELECOM ParisTech – INFRES – CNRS LTCI - UMR 5141 – Paris, France

Email: {karel.devogeleer,gerard.memmi}@telecom-paristech.fr

†MINES ParisTech, France

Email: {pierre.jouvelot,fabien.coelho}@mines-paristech.fr

Abstract—We introduce and experimentally validate a new macro-level model of the CPU temperature/power relationship within nanometer-scale application processors or system-on-chips. By adopting a holistic view, this model is able to take into account many of the physical effects that occur within such systems. Together with two algorithms described in the paper, our results can be used, for instance by engineers designing power or thermal management units, to cancel the temperature-induced bias on power measurements. This will help them gather temperature-neutral power data while running multiple instance of their benchmarks. Also power requirements and system failure rates can be decreased by controlling the CPU’s thermal behavior.

Even though it is usually assumed that the temperature/power relationship is exponentially related, there is however a lack of publicly available physical temperature/power measurements to back up this assumption, something our paper corrects. Via measurements on two pertinent platforms sporting nanometer-scale application processors, we show that the power/temperature relationship is indeed very likely exponential over a 20°C to 85°C temperature range. Our data suggest that, for application processors operating between 20°C and 50°C, a quadratic model is still accurate and a linear approximation is acceptable.

I. INTRODUCTION

Since the dawn of Integrated Circuits (ICs), it is known that the currents flowing through the ICs produce heat dissipation proportional to I^2R , where R is the resistance and I the electric current. The first law of thermodynamics states that in steady operation the energy input of a system is equal to the energy output of the system. Thus, in the absence of other energy interactions, most energy leaves an IC in the form of heat, resulting from currents flowing through electrical elements [1]. Therefore the IC’s heat dissipation is proportional to its power consumption. Moreover the IC will exhibit transient thermal behavior, where the time-frame depends on its heat capacity. The transient thermal behavior is more lasting for systems with larger heat capacities than systems with smaller heat capacities. Such systems can be thought of as RC-circuits, i.e., a low-pass filter, where the temperature of the system is proportional to the voltage over the capacitors [2]. In reality the (transient) thermal behavior is more complicated, because the resistance R and the current I are known to depend on the IC’s temperature T , which is non-

uniform over the IC and time-dependent. Generally speaking, it is shown that the power dissipation grows super-linear with the temperature of the system.

The IC’s heat generation process is sometimes also referred to as *internal heat conversion*. It is the *temperature/power relationship*, i.e., the relationship between the internal heat conversion and the temperature, that is the subject of this work. In particular we study the thermal behavior of three application processors for embedded systems such as smartphones, notebooks, or tablets.

Elevated temperatures have adverse effects on ICs. The reliability of electronic products can be influenced by spatial or temporal gradients, or absolute temperatures [3]. The Mean Time To Failure (MTTF) of electronic equipment increases exponentially with temperature; possible causes of failure are electron migration, chemical reactions, dielectric breakdown, or creep in the bonding materials [1]. For safety reasons, the system’s temperature is also limited. Smartphones are often thermally capped around 50°C so that its users don’t burn any body parts, but also to maximize battery life and up-time. Such applications employ Thermal Management Units (TMUs), which are able to throttle or scale the system so that stringent thermal constraints are met. Complex Central Processing Units (CPUs) may employ hard-wired TMUs, e.g., some Intel chip sets [4], [5], whereas TMU software implementations are frequently seen in embedded devices. For TMUs, it is important to understand the transient thermal behavior of systems. The transient thermal behavior of actively cooled systems can be well described via an exponential relationship. But, for passively cooled systems such as smartphones, wireless sensors, appliances or vehicles, the exponential assumption does not hold. Passively cooled systems rely on cooling via conduction, natural convection, radiation and the TMU, which exhibit non-linear properties. To be optimally effective, TMUs must therefore have an adequate understanding of the system. This includes knowledge about its transient thermal behavior, the internal heat conversion and its temperature dependency.

There are several factors that affect the internal heat conversion dependency on temperature. Some manifest at the micro-level; others result from macro-level effects. The most

known contributor to the temperature dependence of internal heat conversion are the *transistor leakage currents*. Leakage currents effects are inherent to silicon-based metal-oxide semiconductor field-effect transistors (MOSFETs) with which CPUs are built. It is well known that these currents are temperature-dependent [6]. On a macro level, a CPU is fed by a voltage regulator which also contains a cohort of transistors and other electrical elements. Given the temperature-dependent behavior of transistors, the voltage regulator will be temperature-dependent as well. Moreover, the physical properties such as the *electrical resistance* and *thermal resistance* of the materials that compose the CPU are themselves temperature-dependent. It is the combined effect of such phenomena that result in the non-linear thermal behavior of systems. Many theoretical studies focus solely on the effects of leakage currents, a priori neglecting any other sources responsible for temperature-dependent behavior.

From a measurement perspective, it is also important to understand the implications of the internal heat conversion temperature dependency. The reproducibility of accurate power measurements are challenging by virtue of the transient thermal behavior of ICs. More specifically, for a fixed benchmark, a CPU power measurement will yield different values at different CPU temperatures. For the sake of accuracy and fair comparison between different power measurements, it is thus of vital importance to control or cancel the effects of transient and static thermal behaviors.

The main contributions of this paper are:

- experimental evidence that the aggregated behavior of the temperature/power relationship is very likely exponential, while, for small temperature variations, quadratic and linear approximations may be adequate;
- a new model that estimates the influence of temperature on an application processor's CPU power consumption, for a given frequency and active core count;
- a simple method to remove the temperature bias from a power measurement trace.

The rest of the paper is organized as follows. Section II elaborates on the processes that may influence the temperature-dependent behavior of the internal heat conversion. In Section III, examples of temperature/data traces found in the literature are listed; they illustrate the internal heat conversion. Section IV describes the testbed and measurement methodology we used to gauge the internal heat conversion in our own measurements. Section V presents the temperature-power traces we collected and their analyses, plus a generalized power model. Section VI presents a real-life application of the results from the previous Section. Section VII sheds some light upon possible future work, mainly on how to improve the accuracy of the measurements. We conclude in Section VIII.

II. CONTRIBUTORS TO TEMPERATURE FLUCTUATIONS

The *total power consumption* P_{cpu} of a CPU is linked to all the currents that flow within the CPU accounted for by different physical processes [7]. Firstly, the *dynamic power* P_{dyn} is the power used to charge and discharge the capacities

that drive the logic gates within the CPU. Each time the CPU changes its state, i.e., gates are toggled, energy is required or flushed to maintain this new state. Moreover, during the state changes of the logic gates, the transistors inside may conduct simultaneously for a very brief moment, shorting the supply to ground. Currents flowing as part of this process contribute to the *short-circuit power* consumption P_{short} . Also, leakage currents flow through the transistors as a result of their non-ideal behavior resulting in a leakage power P_{leak} . The total CPU power consumption can then be thought of as the sum of the mentioned processes, whence

$$P_{\text{cpu}} = P_{\text{dyn}} + P_{\text{short}} + P_{\text{leak}}. \quad (1)$$

P_{leak} is known to be temperature-dependent and well defined for an isolated transistor. P_{dyn} and P_{short} are related to changing physical material properties resulting from temperature fluctuations. Furthermore, P_{cpu} is itself temperature-sensitive as the voltage supply level may be temperature-dependent. The total internal heat conversion is the combined effect of the mentioned processes whose relations are not totally clear, and moreover could lag in time. Therefore, further on, we approach the modeling of the temperature/power relationship at a macro level such that the aggregated behavior of the mentioned, and tentatively other, physical processes are captured.

A. Leakage Currents

Among the multiple sources of leakage in MOSFET transistors, the sub-threshold leakage current, gate leakage and band-to-band tunneling (BTBT) dominate the others for sub-100nm technologies [8], [9]. Leakage current models, e.g., as incorporated in the Berkeley Short-channel IGFET Model (BSIM) [6], are accurate, nevertheless complex, since they depend on multiple parameters. Detailed knowledge of the transistors are necessary to assess the precise magnitude of the leakage currents, e.g., dimensions, materials and terminal voltages. Even more, when transistors are stacked, e.g., in logic-gates, leakage currents may be amplified throughout the stack [9]. As a CPU changes state each clock cycle, keeping track of the exact leakage current over time may practically be a daunting task.

As the physical dimensions of transistors shrink each generation and materials used are optimized, one should be careful to assess the magnitude of leakage currents based on past research. It has been shown that, for example, gate leakage becomes more prominent when transistor dimensions shrink [10]. Mostly sub-threshold leakage currents are accounted for in previous research, which was not necessarily a wrong assumption for the larger transistor sizes studied in the past. However, as a result of composed leakage current effects, we are poised to develop a model that captures this aggregated behavior over time and its temperature dependency, on a macro level for all transistors in the CPU.

Often course-grained models are inspired by the intrinsic behavior of a single transistor's leakage current. Table I provides an overview of models found in the literature. Liao et al. [12] stated that for their 65 nm benchmark the sub-threshold

TABLE I
LITERATURE MODELS CAPTURING THE COARSE-GRAINED BEHAVIOR OF
LEAKAGE CURRENTS. a_* ARE SCALARS, T IS THE CPU TEMPERATURE.

AUTHORS	MODEL
Su [11]	$P_{\text{leak}} \propto a_2 T^2 + a_1 T + a_0$
Liao [12]	$P_{\text{leak}} \propto a_1 T^2 e^{a_0/T}$
Liao [13]	$P_{\text{leak}} \propto a_2 e^{-a_0/(T-a_1)}$
Liu [14]	$P_{\text{leak}} \propto a_2 + a_2(T - a_1)$
Ferré-Sinha [15] [16]	$P_{\text{leak}} \propto a_1 e^{a_0/T}$
Skadron [17]	$P_{\text{leak}} \propto a_1 T^2 e^{-a_0/T}$
Skadron [18]	$P_{\text{leak}} \propto a_2(1 - e^{a_1/T})e^{a_0/T}$

and gate leakages dominate the leakage process. Only the former is temperature-dependent, the authors claim. In another paper by Liao et al. [13] a power consumption model for adders was presented. The temperature-dependent part of the power model looks slightly different from their previous work. Skadron et al. [17] deducted a relationship between the leakage power P_{leak} and dynamic power P_{dyn} based on International Technology Roadmap for Semiconductors (ITRS) power traces. It can be observed that their equation is based on the sub-threshold leakage current. In another publication, Skadron et al. [18] adopted the exact formulation of the sub-threshold leakage currents for the transistor in the *off* state. Liu et al. [14] put forward a linearized leakage current equation based again on the gate and threshold leakage current, which was studied via SPICE simulations. Yet, in their humble attempt to model the thermal behavior of leakage currents with finite elements, they forgot to account for the inflating power consumption due to leakage currents. Su et al. [11] modeled the leakage currents of so called *standard cells* using SPICE and custom thermal simulations. The authors identified a satisfactory quadratic correlation between temperature and leakage currents. Ferré and Figueras [15] and also Sinha and Chandrakasan [16], based on the sub-threshold leakage current, assumed a pure exponential relationship between temperature and leakage.

Most authors assert that they were able to model the leakage currents adequately on their dedicated testbed. This suggests that leakage currents may very well be application-specific, i.e., for given transistor dimensions and materials etc. Measuring the leakage current on a real testbed is not a straightforward task; therefore simulations are often employed to quantify its magnitude. Yet, we can get a glimpse of its behavior by adjusting temperature levels.

B. Voltage Regulators

Each CPU has a voltage regulator that supplies the CPU with a constant voltage supply. The voltage regulators of Dynamic Voltage and Frequency Scaling (DVFS)-enabled CPUs can alter the magnitude of the supplied voltage on demand, though with a small transition delay. A voltage regulator is built up from capacitors, inductors and transistors. Therefore the voltage regulator will also supply a voltage where its magnitude depends on the temperature. The resulting macro-level effect is that, for a fixed resistance, the current supplied to a resistor will increase and augment the internal heat conversion. For example, via Ohm's law, we calculate that, if a 1.5 voltage

drop over a 1 k Ω resistor increases by 1%, the resistor's power dissipation will increase by 2%.

The temperature sensitivity of the voltage regulator is usually listed in its datasheet. The S2MPS11 voltage regulator of our ODROID testbed (defined in the sequel) is unfortunately not at our disposal. But we can estimate its temperature drift via the onboard INA231 voltage sensor. We observed in the most extreme case¹ a voltage rise of 1.25 V to 1.265 V between 30°C and 90°C for the A15 processor, which corresponds to a 1.2% voltage increase. The maximum gain error for the INA231 is listed to be 0.5%. This leaves us with an estimated voltage regulator temperature drift of around 0.8%. For the more energy-efficient A7 processor, in idle mode a 0.25% rise was noted in the most extreme case. The quantization noise and gain error, however, render the latter observation unreliable. Regardless of the large measurement errors, we have an indication that temperature may affect the voltage regulator's output.

In general, the voltage regulator may not always be exposed to the full temperature swings stemming from switching logic inside the CPU. This depends on the relative distance of the voltage regulator to the CPU. As a result, during peak CPU activity, escalating internal heat conversion, resulting from leakage current swells, will kick in faster than the increased dissipation from the inflating voltage supply. This is a result of the finite propagation time of heat between the CPU's logic and the voltage regulator. As an illustration, in both our testbeds the voltage regulator (S2MPS11 and MAX8997) is located about 1 cm away from the actual application processor. Practically this implies that the transient thermal behaviors of a CPU when the whole system is heated homogeneously, e.g., while being exposed to the sun, and when the temperature rise in the CPU emanates from the execution of a job will look different.

C. Physical Properties of the CPU

Physical properties of the materials that constitute the CPU and printed circuit board (PCB) are temperature-dependent, including the electrical resistance and thermal diffusivity. For example the resistivity of copper and aluminum increases about 12% between 0°C and 50°C. As an other practical illustration, note that a thick film resistor's, e.g. surface mount device (SMD), electrical resistivity decreases 1% over 50°C. The electrical resistivity of semiconductors typically decreases with rising temperatures. Similarly, the thermal conductivity of both copper and aluminum changes about 0.9% between 0°C and 50°C.

Even though these physical properties are temperature-dependent, they probably have a small influence on the temperature dependence of the internal heat conversion.

III. TEMPERATURE/POWER MODELS IN THE LITERATURE

With the objective of reaching adequate performance within temperature constraints, DVFS controllers may employ temperature/power models. Also for TMUs it may be useful to under-

¹All CPU cores active at maximum frequency.

stand the thermal behavior of the temperature/power relationship. Computer energy consumption decompositions account often for the leakage currents where the temperature/power dependency is referenced. A summary of temperature/power relationship models found in the literature is listed below.

Weissel and Bellosa [19] developed a TMU for data center computers. Based on a handful of temperature/power measurements in a limited temperature range (35°C to 60°C), they assumed the temperature/power relationship to be quadratic, quasi linear. The accuracy of the fitting is however questionable. Hanumaiah and Vrudhula [20] developed a DVFS controller for systems with hard real-time and temperature constraints. They employ a linearized version of the exponential temperature/power assumption, which was based on the BSIM leakage current models. The authors also assume that the power increases linearly with the supply voltage. The temperature in their experiments ranged between 35°C and 110°C. While studying the thermal response to DVFS of an Intel Pentium M processor, Hansom et al. [21] assumed a linear relationship between power and temperature. The temperature ranged between 20°C and 55°C in their experiments. Sinha and Chandrakasan [16] decomposed the energy consumption of a StrongARM platform. Based on the BSIM definition of the sub-threshold leakage current, they proposed an exponential relationship to represent the leakage current. The temperature is implicitly referenced in the denominator of the natural exponent; other than that the temperature is not mentioned. Liao et al. [12], in their simulations, assess a CPU's performance. The authors also assume an exponential behavior based on the sub-threshold leakage current. The temperature in their experiments ranges between 65°C and 110°C. Singh et al. [22] attempted to model an AMD CPU's power consumption based on a subset of performance counters. Temperature/power traces are shown but only with relative figures. Their traces show that during the execution of some benchmarks the CPU's temperature inflates about 20%, resulting in a 10% power increase. With a bit of good-will, a super linear relationship can be identified. Ikebuchi et al. [23] show temperature/power traces for their Geyser-1 MIPS CPU. The temperature/power relationship, measured between 20°C and 80°C, shows a clear exponential relationship. The authors also show that with the help of power-gating the effects of leakage currents on power consumption can be diminished.

The works listed above show temperature/power traces for at most three benchmarks and for specific CPU settings, mostly for illustrative purposes. Usually high-performance MIPS processors are targeted as the objects of study, as part of large server farms. Based on elaborate measurements described further on, we identify an experimental temperature/power relationship for different CPU configurations and loads for our application processors. Such application processors are expected to function in embedded systems, e.g., smartphones, appliances, vehicles or smart sensors.

IV. TESTBED

We used the following two platforms to collect temperature/power traces. The first is a Samsung Galaxy S2 sporting the Samsung Exynos 4 Systems-on-Chip (SoC) 45 nm dual-core, and the second, a Hardkernel ODROID XU+E featuring the Samsung Exynos 5 SoC 28 nm quad-core. The Galaxy includes an A9 Cortex processor, whereas the ODROID has both an A7 and an A15 Cortex processor on the same die. The two platforms were running a custom compiled Linux kernel. The frequency scaling governor was set to operate in *userspace* mode to prevent frequency and voltage scaling on-the-fly.

The ODROID has onboard power sensors with an accuracy around 1.25 mW. To measure the power consumption of the Galaxy we replaced its battery with a power supply (Monsoon Power Monitor) that samples the power with about 1 mW accuracy. The temperature on both platforms was measured via onboard temperature sensors with a 1°C accuracy. Power and temperature samples were collected at a rate of 5 Hz. We applied forced cooling and forced heating to the SoCs packaging (including the CPU) to force its temperature up and down.

During the trace recording a constant load is applied to one or more cores of the CPUs. The Galaxy was loaded with 4096 kB bit-reverse calculations. We used the Gold-Rader implementation of the bit-reverse algorithm, part of the ubiquitous Fast Fourier Transformation (FFT) algorithm, which rearranges deterministically elements in an array. The ODROID spun over the square root function from the default math library. The root calculations were forked up to four times to assess the temperature/power impact of the four cores in the A7 and A15 processors; the inactive cores were not hot-plugged. On the A9 platform we only enabled one of the two cores; the other core was unplugged. Some other CPU peripherals were disabled, including the screen and camera, to minimize noise in the power measurements. It must be noted that the benchmarks ran on top of an Operating System (OS), so there must be some power accounted to the system's overhead.

V. TEMPERATURE/POWER MODELING

Because of the temperature dependency of some currents flowing through the CPU, the CPU power consumption will inflate for increasing silicon temperatures. We describe our experiments and the temperature/power model we deduced from them.

A. Experiments

We artificially swept the temperature between 25°C and 85°C for the A7 and A15 processors; for the A9 the temperature was swept between 25°C and 55°C. We measured the power consumption and temperature of the A7 between 250 MHz and 600 MHz, the A9 between 200 MHz and 1.6 GHz, and the A15 between 0.8 GHz and 1.6 GHz. Excerpts of the traces are shown in Figure 1 and Figure 2.

We fitted all the traces with three types of curves to assess

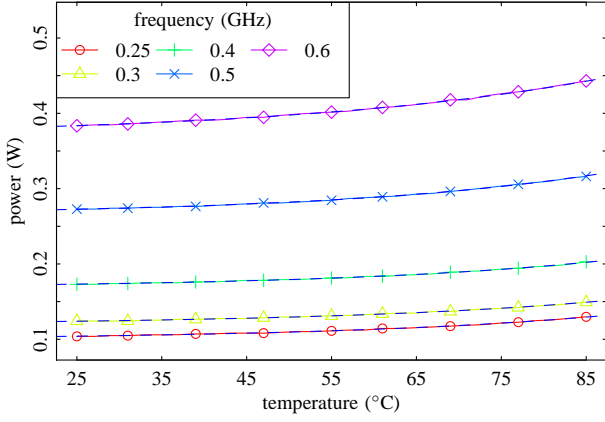


Fig. 1. Temperature/power traces for the A7 processor with three active cores at different frequencies. The dashed blue lines are the fitted exponential curves as described in Equation 4.

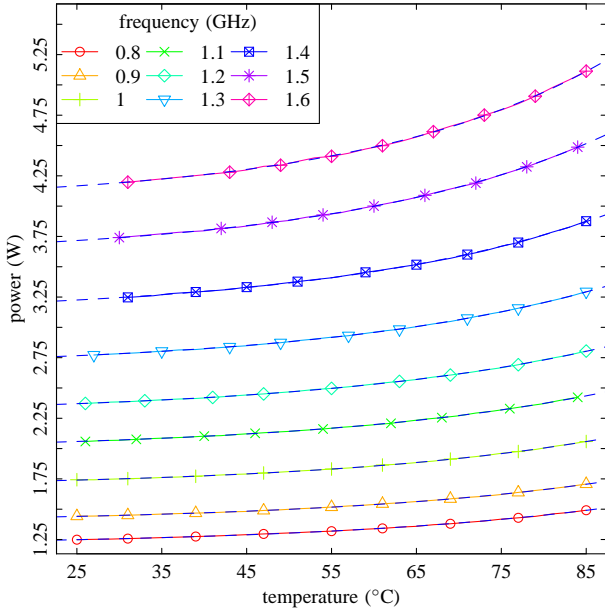


Fig. 2. Temperature/power traces for the A15 processor with four active cores at different frequencies. The dashed blue lines are the fitted exponential curves as described in Equation 4.

their applicability:

1) linear curve:

$$P = a_1 T + a_0; \quad (2)$$

2) quadratic curve:

$$P = a_2 T^2 + a_1 T + a_0; \quad (3)$$

3) exponential curve:

$$P = e^{(T-a_1)/a_2} + a_0. \quad (4)$$

where a_* are scalars, to be defined via fitting. Figures 3a and 3b show single temperature/power traces for the A7 and A15 processors, respectively. Almost all traces look similar to these two examples; therefore, and for space reasons, we do not show all curves and present aggregated fitting errors. Aggregated fitting errors are given in Table II for fitting over

TABLE II
AGGREGATED TEMPERATURE/POWER LINEAR, QUADRATIC (QUAD) AND EXPONENTIAL (EXPO) FITTING ERRORS OVER THE TEMPERATURE RANGE 25°C TO 85°C FOR THE A7 AND A15 PROCESSORS (PR) AND A GIVEN ACTIVE CORE COUNT (# CO).

PR	# CO	LINEAR	QUAD	EXPO
A7	1	0.190058	0.048509	0.039109
A7	2	0.126700	0.030710	0.023524
A7	3	0.111039	0.026028	0.016972
A7	4	0.103021	0.023661	0.014973
A15	1	0.120828	0.021422	0.008501
A15	2	0.098198	0.017627	0.007960
A15	3	0.085662	0.014614	0.006146
A15	4	0.084010	0.014459	0.005787

TABLE III
AGGREGATED TEMPERATURE/POWER LINEAR, QUADRATIC (QUAD) AND EXPONENTIAL (EXPO) FITTING ERRORS (10^{-3}) OVER THE TEMPERATURE RANGE 25°C TO 55°C FOR THE A7 AND A15 PROCESSORS (PR) AND A GIVEN ACTIVE CORE COUNT (# CO).

PR	# CO	LINEAR	QUAD	EXPO
A9	1	0.197413	0.147489	0.158332
A15	1	9.519750	4.648533	4.590021
A15	2	8.536068	4.558667	4.596187
A15	3	6.327271	3.244339	3.206688
A15	4	6.318451	3.000202	2.96603

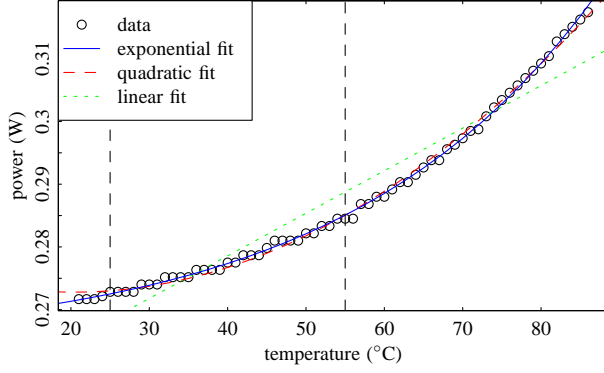
the 25°C to 85°C temperature range. The fitting errors are aggregated over all the traces measured with the same active core count. The fitting errors were computed as follows:

$$\text{error} = \sqrt{\sum_i \left(\frac{\tilde{y}_i - y_i}{y_i} \right)^2}, \quad (5)$$

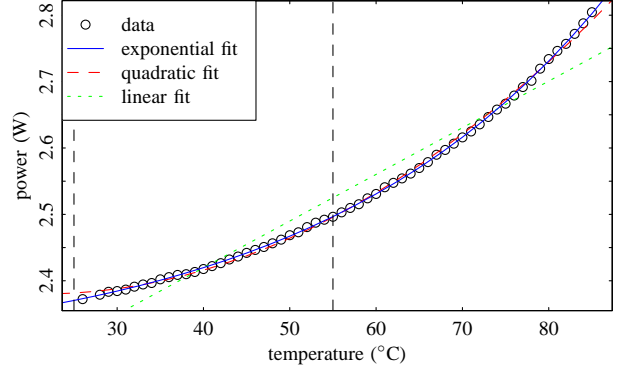
where y are the measured values and \tilde{y} , the model.

We observe that the sum of errors for the quadratic case are on the average 2.5 times larger than the exponential fit errors, and the linear fit errors are about six times larger than the quadratic. The p -values of the *sign test* between the three models on the A7 and A15 stay well below the 0.01 significance level, confirming that the exponential is a significantly better fit than the quadratic, while the latter is significantly better than the linear fit. Thus, the exponential fit would be the most representative of the three proposals. Figures 3a and 3b back up this observation. Indeed, the exponential fit seems to follow the measurements very well. The quadratic curve overestimates the power for the lower temperatures but performs very well for larger temperatures. The linear curve does not adequately represent the temperature/power relationship in this temperature range compared to the other two proposals.

Let's see how the curves behave in the pertinent temperature range between 25°C and 55°C. Aggregated errors are given in Table III. Due to the quantization noise within this temperature span, the fitting for the A7 doesn't always converge properly. This hampers the fitting process and renders the results unreliable; therefore we omit its analysis here. For the A15 processor, we observe that the competitive advantage of



(a) A7 processor - 500 MHz - #pr=3



(b) A15 processor - 1.2 GHz - #pr=4

Fig. 3. Temperature/power relationship as measured on the A7 and A15 processors for three and four active cores (#pr), respectively. The traces for the other processors and configurations show similar behavior. Note the quantization noise in the case of the A7 processor, most prevalent at low temperatures.

the exponential curve has shrunk. The *sign test* significantly favors the quadratic curve over the linear curve. The quadratic curve in the 25°C–55°C temperature range is, however, as good as the exponential curve based on the same sign test. In the case of the A9 processor, based on the aggregated errors, the quadratic curve presents itself as the best fit, but it is as good as the exponential according to the sign test with a 0.01 significance level. Nevertheless, the linear curve is also a good match compared to the quadratic and the exponential. We must note, however, that the A9 traces suffer from the so-called *distant sensor syndrome*, described in Section VII.

Even though the curves are most likely exponential, in this limited temperature range the quadratic curve performs as well as the exponential curve. The performance of the linear curve is also acceptable. This is a positive conclusion for TMUs and previous research that assumed a linear or quadratic relationship between temperature and power. Analytical derivations can be notably simplified by virtue of said assumptions.

B. General Temperature/Power Model and its Error Analysis

For diverse purposes such as simulations, among others, it is useful to define the scalars a_* in Equation 4 for arbitrary frequencies and active core count. From analyzing the A7 and A15 traces it appears that a_2 is seemingly constant for all measurement data. Observing the fitted values for a_1 reveals that these are linearly correlated with frequency and active core count. For the a_0 case, the values are quadratically correlated with the frequency and linearly with the active core count. Moreover, the lines linking the a_2 values for a fixed frequency extrapolated seem to converge towards a single point on the abscissa. Thus we suggest to use the following expressions for a_0 and a_1 :

$$\begin{aligned} g_s &= m_1 + m_2 f + m_3 f^2 \\ g_o &= g_s / m_4 \\ a_0 &= g_s c + g_o \\ a_1 &= m_5 f + m_6 + (5 - c) m_7, \end{aligned} \quad (6)$$

$$(7)$$

procedure POWER.A7(T, f, c)

```

 $g_s \leftarrow 0.028 - 0.093f + 0.371f^2$ 
 $g_o \leftarrow g_s / 2.202$ 
 $a_0 \leftarrow g_s c + g_o$ 
 $a_1 \leftarrow -38.242f + 187.668 + (5 - c)8.430$ 
 $a_2 \leftarrow 33.105$ 
return  $\exp((T - a_1)/a_2) + a_0$ 

```

end procedure

Fig. 4. Algorithm to compute an A7 power estimate given the temperature T (°C), CPU frequency f (GHz) and active core count c (1:4).

procedure POWER.A15(T, f, c)

```

 $g_s \leftarrow 0.220 - 0.315f + 0.467f^2$ 
 $g_o \leftarrow g_s / 2.202$ 
 $a_0 \leftarrow g_s c + g_o$ 
 $a_1 \leftarrow -56.652f + 165.896 + (5 - c)8.430$ 
 $a_2 \leftarrow 33.105$ 
return  $\exp((T - a_1)/a_2) + a_0$ 

```

end procedure

Fig. 5. Algorithm to compute an A15 power estimate given the temperature T (°C), CPU frequency f (GHz), and active core count c (1:4).

where f is the CPU frequency, c , the active core count and m_* , case-specific scalars. For both processors we observed that m_4 and m_7 are approximately equal. We have also tried to see whether the CPU supply voltage may be correlated with the a_* scalars; however, we haven't identified a satisfactory correlation.

A prototype implementation of the power models for the A7 and A15 is given in Figure 4 and Figure 5, respectively, which can be copy-pasted directly into any simulation software. Based on the collected traces, the A15 power model above shows a median error of 1.19% and a maximum error of 7.11%; for the A7 processor the median error is 2.89% and the maximum is 8.31%. In absolute terms, both models deviate about equally from the measurements, but as the A7 consumes less power its relative error is larger. The presented errors are not negligible. After analyzing our data we have

identified so far three sources that introduce errors/noise: the initial temperature conditions that vary for each trace, the temperature sensor noise and the non-uniform temperature gradient while heating.

VI. TEMPERATURE-BIAS CANCELLATION IN POWER CONSUMPTION MEASUREMENTS

In the previous section we have shown that the temperature/power relationship is very likely exponential. A linear or quadratic relationship is adequate as well for a limited temperature range. In this section we provide one application of these results by showing how to improve power measurement accuracy by canceling the inflations stemming from temperature fluctuations.

Figure 6 shows three examples of actual power measurement traces; the temperature was also recorded. Our goal is to convert all measured power samples as if they were measured at a fixed arbitrary *reference temperature*. Based on the temperature/power relationship fittings provided in the previous section, we may assume that there must exist a transformation function (linear/quadratic/exponential) such that the transformed power is constant², i.e., normalized w.r.t. a reference temperature. As a result the power traces as shown in Figure 6 should appear flat after the transformation. A linear relationship between temperature and power ($P = \eta_1 T + \eta_0$) yields the following transformation function, similar to differential approximation:

$$\begin{aligned} P_r - \eta_1 T_r &= P_m - \eta_1 T_m \\ P_r &= P_m + \eta_1 \Delta T, \end{aligned} \quad (8)$$

where P_r is the power at the reference temperature, T_r , the reference temperature, P_m and T_m , the measured power and temperature and $\Delta T = T_r - T_m$. Similarly, a quadratic temperature/relationship ($P = \eta_2 T^2 + \eta_1 T + \eta_0$) yields the following power transformation function:

$$P_r = P_m + \eta_2 (T_r^2 - T_m^2) + \eta_1 \Delta T. \quad (9)$$

To find the optimal transformation function it is not necessary to know the CPU's precise thermal behavior. A linear or quadratic regression between temperature and power of the collected traces suffices to obtain the η_* values. As stated before, this approach is appropriate when the testbed temperature variations are no more than 30°C; otherwise one needs to resort to exponential fits to maintain acceptable accuracy.

Figure 6 shows random irregular power traces and their resulting linear (blue) and quadratic (red) power transformations. As can be observed, the jerky power traces are converted into stable traces, except for the presence of some noise. The temperature noise and inaccuracy is a known problem for TMUs [24]. The most important feature of our power transformation is that the arbitrary distribution is transformed into a symmetric distribution. Table IV shows an overview of the transformation performance for different reference temperatures. The maximum measured power fluctuation (AFL) due

²In practice, due to noisy measurements, the variance of the power measurements is to be minimized.

TABLE IV
PERFORMANCE METRICS OF THE POWER TRANSFORMATION FOR DIFFERENT REFERENCE TEMPERATURES T (°C) AND PROCESSOR CONFIGURATIONS, AS STATED IN FIGURE 6. THE MEASURED MAXIMUM POWER INFLATION IS PROVIDED (AFL), AS WELL AS THE RELATIVE FLUCTUATION (FL) FOR THE LINEAR (L) AND QUADRATIC CASES (Q). THE RATIO BETWEEN THE MEDIAN AND THE MEAN IS ALSO PROVIDED (RAT) TO ASSESS THE MEASUREMENT DISTRIBUTION'S SYMMETRY.

A15 @ 1.3 GHz					
T	AFL	FL-L	FL-Q	RAT-L	RAT-Q
35	1.21	0.297	0.311	$0.753 \cdot 10^{-3}$	$0.403 \cdot 10^{-3}$
38	1.21	0.295	0.310	$0.750 \cdot 10^{-3}$	$0.401 \cdot 10^{-3}$
41	1.21	0.294	0.309	$0.747 \cdot 10^{-3}$	$0.400 \cdot 10^{-3}$
A15 @ 0.9 GHz					
T	AFL	FL-L	FL-Q	RAT-L	RAT-Q
43	1.80	0.231	0.212	$-1.706 \cdot 10^{-3}$	$-2.591 \cdot 10^{-3}$
46	1.80	0.229	0.211	$-1.697 \cdot 10^{-3}$	$-2.579 \cdot 10^{-3}$
49	1.80	0.228	0.210	$-1.688 \cdot 10^{-3}$	$-2.565 \cdot 10^{-3}$
52	1.80	0.227	0.208	$-1.679 \cdot 10^{-3}$	$-2.551 \cdot 10^{-3}$
A7 @ 0.6 GHz					
T	AFL	FL-L	FL-Q	RAT-L	RAT-Q
50	2.99	0.359	0.327	$-0.249 \cdot 10^{-3}$	$8.677 \cdot 10^{-3}$
54	2.99	0.355	0.324	$-0.247 \cdot 10^{-3}$	$8.615 \cdot 10^{-3}$
58	2.99	0.352	0.321	$-0.245 \cdot 10^{-3}$	$8.532 \cdot 10^{-3}$
62	2.99	0.349	0.317	$-0.243 \cdot 10^{-3}$	$8.430 \cdot 10^{-3}$

to varying temperature is shown to be between 1.21% and 3%. The relative power fluctuation is computed as the relative transformed power fluctuation over the measured power fluctuations:

$$FL = \frac{\max(P_r) - \min(P_r)}{\text{median}(P_r)} \bigg/ \frac{\max(P_m) - \min(P_m)}{\text{median}(P_m)}. \quad (10)$$

It can be seen that the power fluctuations are diminished by a factor of about three to four in all cases. This can also be visually verified in Figure 6. Moreover, the resulting transformation's distribution is quasi symmetric. This is shown with the RAT metric in Table IV, which represents the departure of the median from the mean. If the median differs significantly from the mean, then the distribution is not symmetric. We see however that in all cases the median and mean are very close to each other, indicating that the transformation produces a symmetric distribution. Now that the power is converted as if it were measured at a reference temperature, statistical methods have a more practical meaning. Statistical methods applied directly to the measured power will produce estimates that are inflated based on the arbitrary distribution of the measured power samples.

We note that it is advisable to choose a reference temperature within the measured temperature range, preferably not too close to the extremities, to minimize transformation errors.

VII. FUTURE WORK

Enhancing the accuracy of the presented temperature/power models should be the main objective of future work. A more profound study of the following observations, among others, is necessary. What is the impact of *temperature gradients* on the temperature/power behavior? How do the *initial conditions* of temperature and power affect the temperature/power behavior? How is the *accuracy of the temperature measurements* affected

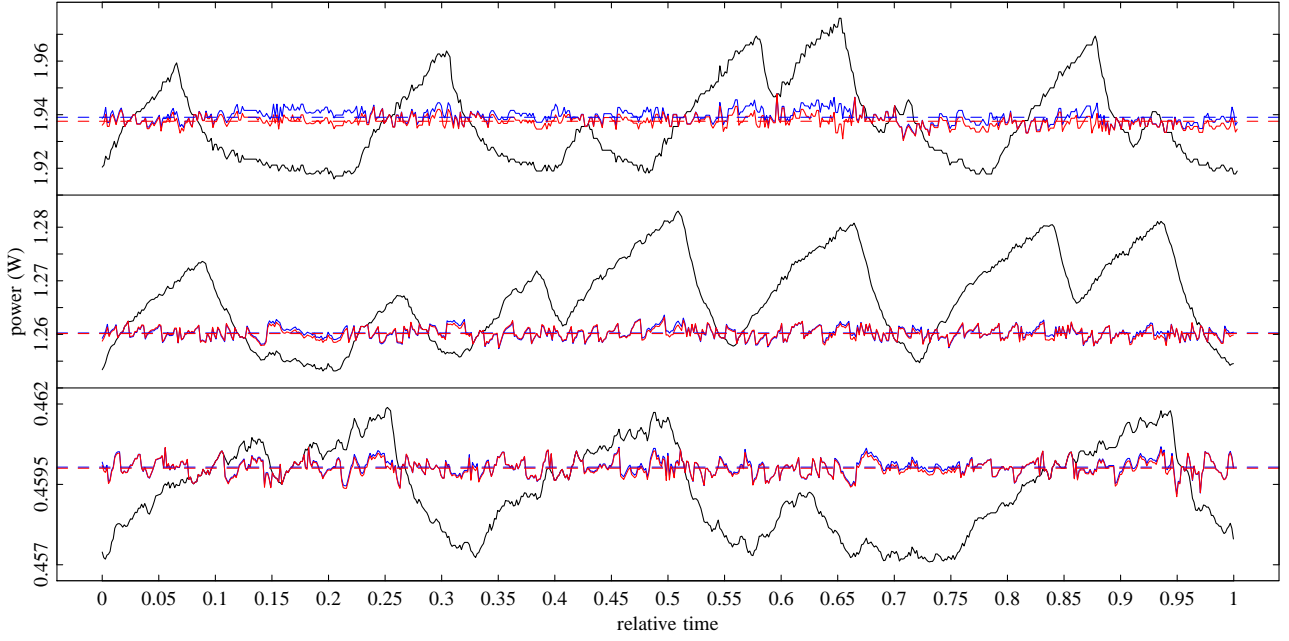


Fig. 6. Power/time traces (black) of (top) the A15 running at 1.3GHz with 3 active cores, (middle) the A15 running at 0.9GHz and (bottom) the A7 running at 0.6GHz with 4 active cores. The blue and red lines are the transformed power traces in the case of linear and quadratic temperature/power approximations, respectively. A random reference temperature was chosen at (from top to bottom) 40°C, 46°C and 55°C.

when the temperature sensor is not located at the temperature hotspots? How can the *temperature sensor accuracy* be improved?

The effects of temperature gradients and initial conditions can be assessed by controlling the testbed more precisely. However, some parameters can only be controlled within certain limits, which poses great challenges for testbeds containing retail devices. Also, the accurate temperature sensors are not always available and often lack calibration facilities.

Previously it was shown that the accuracy of temperature estimates depends, among other factors, upon the distance of the temperature sensor to the temperature hotspots in the processor [24]. Being located away from heat hotspots, the temperature sensor will not only provide an underestimation of the temperature, but the temperature measurements will also lag in time as a result of finite propagation delays, an issue referred to as the *distant sensor syndrome*. Such propagation delays may have some serious implications in extreme cases when the temperature sensors are off-board. This will have an impact on the measured temperature/power relationship. Indeed, there are generally two types of curves found in the literature, as shown in Figure 7. First, the traces where the temperature/power curve bends downwards, as a result of the temperature lag over the power trace (blue line). Examples of such measurements are the ones by Weissel and Bellosa [19], and also our A9 measurements. Measurements of the other type are traces where the temperature sensor is relatively close to the temperature hotspots; these also include results from simulations (green line). Our traces, and the majority of the literature cited before, predict the upwards bending of the curve.

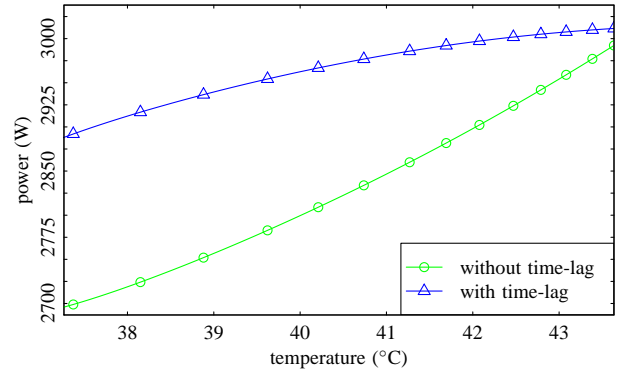


Fig. 7. Theoretical derivations predict monotonically increasing temperature/power curves, indicated by the green line. Frequently, monotonically decreasing curves are measured (top line) as a result of heat propagation delays between temperature hotspots and the temperature sensor. The bottom line was obtained by applying a transformation function to the top line.

Let's assume that, for the downward bending curves, there exist a transformations function named $B(t)$. The transformation function $B(t)$ transforms a temperature measurement of the distant temperature sensor into a measurement as it would have appeared to be at the hotspot: $T_{\text{CPU}}(t) = B(t) \cdot T_{\text{sensor}}(t)$.

A first order approximation to $B(t)$ could be constructed as follows. Let's assume that our temperature sensor is located at a finite distance ($x = a$) in a finite one-dimensional slab. A heat source with temperature T_i is applied to the surface (at $x = 0$ and $t = 0$) of the semi-finite slab, with initial conditions T_∞ for $x \rightarrow \infty$ at $t = 0$. The heat propagation in this case is described by Fourier's law of heat conduction. The well-known solution for this setup is the error function $\theta(x, t) = \text{erf}(x/\sqrt{4\alpha t})$, where α is the thermal diffusivity of

the system and θ the relative temperature change [1]. A more accurate approach would be to model the system in a two-dimensional space with additional thermal resistors to account for the multiple materials that the heat encounters when propagating from the temperature hotspot to the temperature sensor. Assuming that the temperature at the CPU hotspot increases exponentially, a first order approximation to the transformation function $B(t)$ can then be defined as

$$B(t) = \frac{T_{\text{CPU}}}{T_{\text{sensor}}} = \frac{(T_{\infty} - T_i) (1 - \exp(-t/b)) + T_i}{(T_{\infty} - T_i) \operatorname{erf}(a/\sqrt{4\alpha t}) + T_i}, \quad (11)$$

where α , a and b are case-specific scalars.

Figure 7 shows the effect of the transformation function on a temperature/power trace with constants $\alpha = 4.125 \cdot 10^{-7}$, $a = 8.25$ and $b = 36.7$. It is observed that the downward bending curve is indeed transformed into an upward bending curve as forecast by the theory.

The step response function $B(t)$ is however of limited practical value. To reconstruct the temperature at the hotspot via a remote temperature sensor one also needs to know the time-dependent load on the system to assess the power consumption. It is a question of acceptable overhead whether reconstructing the temperature is worthwhile pursuing via a transformation function like $B(t)$.

VIII. CONCLUSION

Via experimental data we have shown that the temperature/power relationship for some application processors shows a distinct exponential behavior, which is in line with theoretical foundations. The exponential behavior is affected by, among other factors, temperature-dependent leakage currents, physical properties and the voltage regulator.

A practical model was presented to estimate the CPU power consumption at a given temperature with arbitrary CPU configurations. We believe this model is useful for simulation purposes, although there is room to improve its accuracy.

We have also presented a real-life application where the effects of inflating temperature on power traces were removed. We showed that the proposed technique can be quite effective. The importance to know the whereabouts of the temperature onboard sensors is also pointed out. The distance between temperature sensor and hotspot can significantly influence the shape of the temperature/power relationship.

REFERENCES

- [1] Y. Cengel and A. Ghajar, *Heat and Mass Transfer: Fundamentals and Applications*. McGraw-Hill Education, 2010.
- [2] A. Cohen, F. Finkelstein, A. Mendelson, R. Ronen, and D. Rudoy, "On estimating optimal performance of CPU dynamic thermal management," *IEEE Computer Architecture Letters*, vol. 2, no. 1, p. 6, 2003.
- [3] P. Lall, M. Pecht, and E. Hakim, *Influence of Temperature on Microelectronics and System Reliability: A Physics of Failure Approach*, ser. The electronic packaging series. CRC Press, 1997.
- [4] R. McGowen, C. Poirier, C. Bostak, J. Ignowski, M. Millican, W. Parks, and S. Naffziger, "Power and temperature control on a 90-nm itanium family processor," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 1, pp. 229–237, 2006.
- [5] D. Duarte, G. Geannopoulos, U. Mughal, K. Wong, and G. Taylor, "Temperature sensor design in a high volume manufacturing 65nm CMOS digital process," in *Custom Integrated Circuits Conference, 2007. CICC '07. IEEE*, 2007, pp. 221–224.
- [6] W. Liu, X. Jin, K. Kao, and C. Hu, "BSIM 4.1.0 MOSFET model-user's manual," EECS Dept., Univ. of California, Berkeley, Tech. Rep. UCB/ERL M00/48, 2000.
- [7] K. De Vogeleer, G. Memmi, P. Jouvelot, and F. Coelho, "The Energy/Frequency Convexity Rule: Modeling and experimental validation on mobile devices," in *Proceedings of the 10th Conference on Parallel Processing and Applied Mathematics*. Springer Verlag, Sept 2013.
- [8] A. Agarwal, S. Mukhopadhyay, C. Kim, A. Raychowdhury, and K. Roy, "Leakage power analysis and reduction: models, estimation and tools," *Computers and Digital Techniques, IEEE Proceedings* -, vol. 152, no. 3, pp. 353–368, may 2005.
- [9] Y.-J. Xu, Z.-Y. Luo, X.-W. Li, L.-J. Li, and X.-L. Hong, "Leakage current estimation of CMOS circuit with stack effect," *J. Comput. Sci. Technol.*, vol. 19, no. 5, pp. 708–717, Sep. 2004.
- [10] Z. Liu and V. Kursun, "Leakage power characteristics of dynamic circuits in nanometer cmos technologies," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 53, no. 8, pp. 692–696, 2006.
- [11] H. Su, F. Liu, A. Devgan, E. Acar, and S. Nassif, "Full chip leakage estimation considering power supply and temperature variations," in *Proceedings of the 2003 international symposium on Low power electronics and design*, ser. ISLPED '03. New York, NY, USA: ACM, 2003, pp. 78–83.
- [12] W. Liao, L. He, and K. M. Lepak, "Temperature and supply voltage aware performance and power modeling at microarchitecture level," *Trans. Comp.-Aided Des. Integr. Cir. Sys.*, vol. 24, no. 7, pp. 1042–1053, Nov. 2006.
- [13] W. Liao, J. M. Basile, and L. He, "Leakage power modeling and reduction with data retention," in *Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design*. New York, NY, USA: ACM, 2002, pp. 714–719.
- [14] Y. Liu, R. P. Dick, L. Shang, and H. Yang, "Accurate temperature-dependent integrated circuit leakage power estimation is easy," in *Proceedings of the Conference on Design, Automation and Test in Europe*, ser. DATE '07. San Jose, CA, USA: EDA Consortium, 2007, pp. 1526–1531.
- [15] A. Ferre and J. Figueras, "Characterization of leakage power in CMOS technologies," in *Electronics, Circuits and Systems, 1998 IEEE International Conference on*, vol. 2, 1998, pp. 185–188 vol.2.
- [16] A. Sinha and A. P. Chandrakasan, "Jouletrack: a web based tool for software energy profiling," in *Proceedings of the 38th annual Design Automation Conference*, ser. DAC '01. New York, NY, USA: ACM, 2001, pp. 220–225.
- [17] K. Skadron, M. R. Stan, K. Sankaranarayanan, W. Huang, S. Velusamy, and D. Tarjan, "Temperature-aware microarchitecture: Modeling and implementation," *ACM Trans. Archit. Code Optim.*, vol. 1, no. 1, pp. 94–125, Mar. 2004.
- [18] Y. Zhang, D. Parikh, K. Sankaranarayanan, K. Skadron, and M. Stan, "Hotleakage: A temperature-aware model of subthreshold and gate leakage for architects," Univ. of V., Tech. Rep. CS-2003-05, Mar. 2003.
- [19] A. Weissel and F. Bellosa, "Dynamic thermal management for distributed systems," in *Proceedings of the First Workshop on Temperature-Aware Computer Systems (TACS'04)*, Munich, Germany, Jun. 2004.
- [20] V. Hanumaiah and S. Vrudhula, "Temperature-aware DVFS for hard real-time applications on multicore processors," *IEEE Transactions on Computers*, vol. 61, no. 10, pp. 1484–1494, 2012.
- [21] H. Hanson, S. Keckler, S. Ghiasi, K. Rajamani, F. Rawson, and J. Rubio, "Thermal response to DVFS: analysis with an intel pentium m," in *Low Power Electronics and Design, ACM/IEEE Symposium on*, 2007, pp. 219–224.
- [22] K. Singh, M. Bhaduria, and S. A. McKee, "Real time power estimation and thread scheduling via performance counters," *SIGARCH Comput. Archit. News*, vol. 37, no. 2, pp. 46–55, Jul. 2009.
- [23] D. Ikebuchi, N. Seki, Y. Kojima, M. Kamata, L. Zhao, H. Amano, T. Shirai, S. Koyama, T. Hashida, Y. Umahashi, H. Masuda, K. Usami, S. Takeda, H. Nakamura, M. Namiki, and M. Kondo, "Geyser-1: a MIPS R3000 CPU core with fine-grained run-time power gating," in *ASP-DAC. IEEE*, 2010, pp. 369–370.
- [24] J. Kong, S. W. Chung, and K. Skadron, "Recent thermal management techniques for microprocessors," *ACM Comput. Surv.*, vol. 44, no. 3, pp. 13:1–13:42, Jun. 2012.